SPIKE PRIME LESSONS

By the Creators of EV3Lessons

INTRODUCTION A MICROPYTHON

BY SANJAY AND ARVIND SESHAN

OBJECTIFS DE LA LEÇON

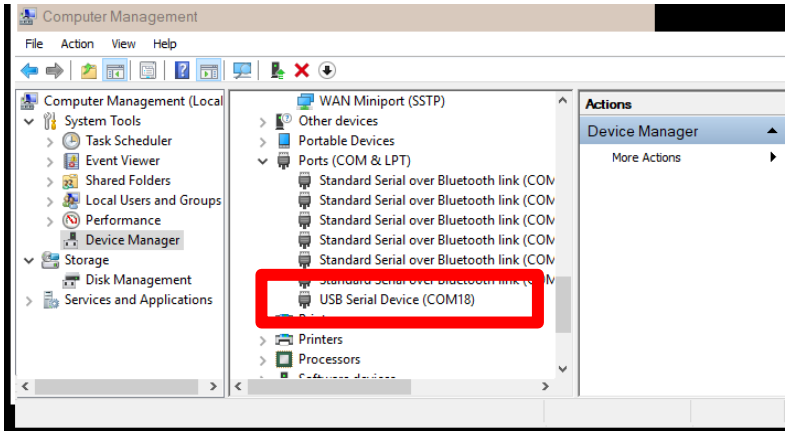
- Apprenez à utiliser MicroPython REPL sur SPIKE Prime

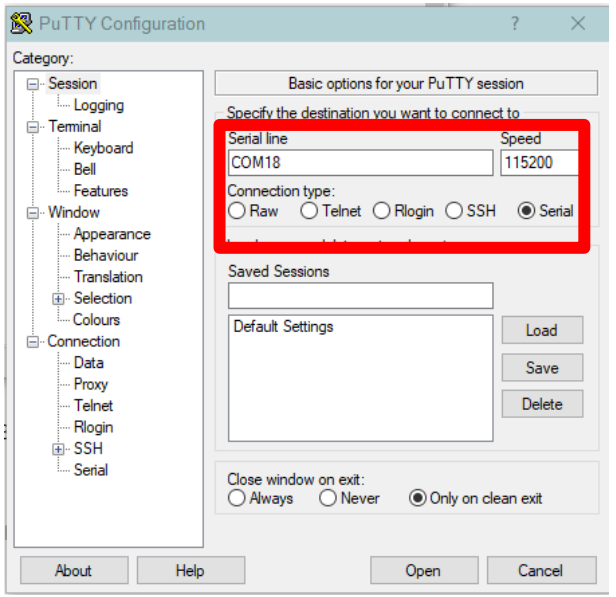
- Pour créer des programmes complets VS. Code à exécuter sur le hub, suivez les instructions à l'adresse https://github.com/sanjayseshan/spikeprime-vscode/wiki

ÉTAPE 1 : SE CONNECTER (WINDOWS)

- Installez l'émulateur de terminal de votre choix
- Exemple : PuTTY https://www.putty.org/

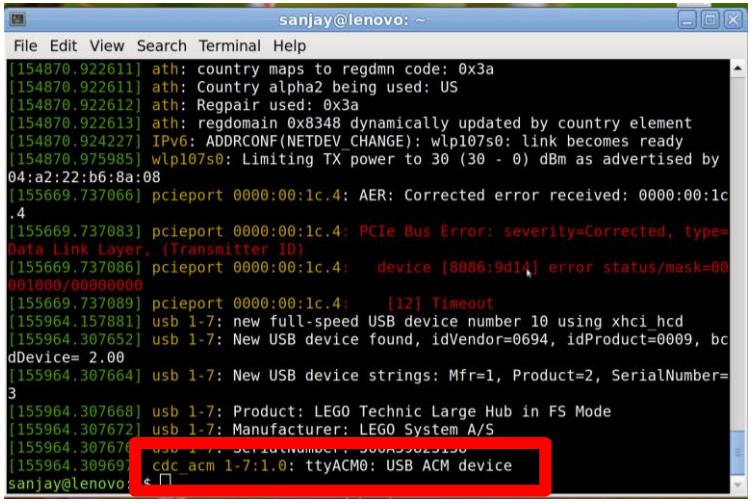
- Assurez-vous que votre logiciel SPIKE Prime ne fonctionne pas
- Connectez le Hub à votre ordinateur via le port USB
- Trouvez le port
  - Sur un PC, regardez dans votre gestionnaire de périphériques (dans le menu Démarrer Outils administratifs de Windows Gestionnaire de périphériques) sous série pour voir quels ports série vous avez connectés
  - Si vous avez plusieurs ports série USB, essayez de vous déconnecter et de vous reconnecter pour voir lequel apparaît
- Connectez-vous au port de droite à 115200 bauds





ÉTAPE 1 : SE CONNECTER (DEBIAN GNU/LINUX)

1. Ouvrez le terminal - Vous pouvez le trouver habituellement sous Applications → System Tools
2. Tapez les commandes suivantes (pour Debian et ses dérivés) :
   1. `sudo apt-get update`
   2. `sudo apt-get install -y screen`
3. Branchez votre Hub et lancez `sudo dmesg`. Une longue liste de messages de log s'affichera. La dernière ligne (ou presque) doit contenir le périphérique USB ACM et l'identifiant similaire à `ttyACM0`. Si vous ne le trouvez pas, cherchez d'abord LEGO Technic Large Hub.
4. Lancez l'écran `sudo /dev/ttyACM0 115200`. Remplacez `ttyACM0` par votre identifiant



ÉTAPE 1 : SE CONNECTER (MAC OS X)

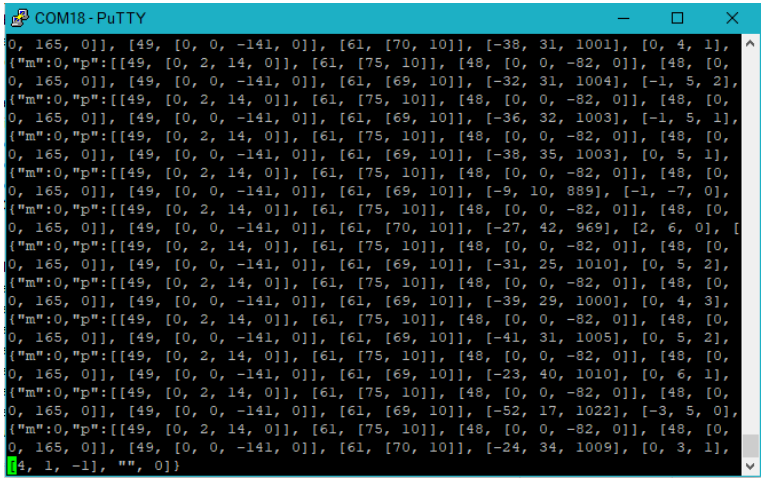
1. Ouvrez le terminal - Vous pouvez le trouver habituellement sous Applications → Utilities
2. Exécutez `ls /dev/ | fgrep usb | fgrep tty` pour trouver le port du hub
3. Exécutez l'écran `/dev/tty.usbmode366A398231381 115200`. Remplacez `tty...` par la sortie de la première commande. S'il y avait plusieurs sorties, essayez-les toutes jusqu'à ce que vous obteniez la bonne.

Vous trouverez ci-dessous un exemple de commandes en cours d'exécution. La sortie de l'ordinateur est en vert, les commandes que vous tapez sont en noir.

```
$ ls /dev | fgrep usb | fgrep tty
tty.usbmodem366A39831234
$ screen /dev/tty.usbmodem366A39831234
```

ÉTAPE 2

- Vous pouvez voir défiler beaucoup de chiffres. C'est le journal des capteurs et des moteurs.
- Hit Control-C
- Vous serez prêt à programmer




```
0, 165, 0]], [49, [0, 0, -141, 0]], [61, [69, 10]], [-43, 38, 995], [0, 4, 2], [
MicroPython v1.9.4-1146-gca9944357 on 2019-10-03; LEGO Technic Large Hub with ST
M32F413xx
Type "help()" for more information.
>>>
```

MODULE HUB

- Le module python "Hub" contient toutes les fonctions/objets clés qui sont nécessaires pour interagir avec le Hub SPIKE Prime.
- Pour accéder à ce module, vous devez d'abord "importer" le module. Tapez "import hub" à l'invite MicroPython

```
>>> import hub
>>>
```

- Une fois que vous avez importé le hub, vous pouvez explorer certaines des interfaces qu'il expose en utilisant l'auto-completion. Tapez "Hub". (assurez-vous d'inclure la période) et appuyez ensuite sur le bouton "Tab"

```
>>> hub.
__class__       __name__        __version__     BT_VCP
Image           USB_VCP         battery         ble
bluetooth       button          display         info
led             motion          port            power_off
sound           status          supervision     temperature
```

COMMANDE D'AIDE

- Le MicroPython sur brique fournit également un outil d'aide limité. Pour accéder à l'aide, tapez "help()"

```
>>> help()
Welcome to MicroPython!

For online help please visit http://micropython.org/help/.

Quick overview of commands for the board:
  hub.info()    -- print some general information
  hub.status()  -- print sensor data

Control commands:
  CTRL-A        -- on a blank line, enter raw REPL mode
  CTRL-B        -- on a blank line, enter normal REPL mode
  CTRL-C        -- interrupt a running program
  CTRL-D        -- on a blank line, do a soft reset of the board
  CTRL-E        -- on a blank line, enter paste mode

For further help on a specific object, type help(obj)
For a list of available modules, type help('modules')
```

AUTRES MODULES/BIBLIOTHÈQUES

- Tapez la commande `help('modules')` (assurez-vous de taper les cotations)

```
>>> help('modules')
__main__          heapq             struct            umachine
_onewire          hub               sys               uos
array             io                time              urandom
binascii          json              ubinascii         ure
builtins          machine           ucollections      uselect
cmath             math              uctypes           ustruct
collections       micropython       uerrno            utime
errno             os                uhashlib          utimeq
firmware          random            uheapq            uzlib
gc                re                uio               zlib
hashlib           select            ujson
Plus any modules on the filesystem
```

- Vous trouverez ici une liste des modules disponibles sur le site SPIKE Prime

AUTRES MODULES/BIBLIOTHÈQUES

- Vous pouvez utiliser la commande d'importation pour charger n'importe quelle bibliothèque que vous trouvez et ensuite utiliser l'autocompletion ou help() pour explorer leurs fonctions

```
>>> import random
>>> help(random)
object <module 'urandom'> is of type module
  __name__ -- urandom
  getrandbits -- <function>
  seed -- <function>
  randrange -- <function>
  randint -- <function>
  choice -- <function>
  random -- <function>
  uniform -- <function>
>>> random.random()
0.711182
>>> random.random()
0.408947
```

DÉFI 1 : BONJOUR LE MONDE

- Imprimez "Hello World" sur votre matrice de lumière
- Quelques étapes clés :
  1. Importez le module Hub
  2. Explorez les composants du Hub pour en trouver un qui contrôle la matrice de lumière (indice : vous voulez "afficher" quelque chose)
  3. Enfin, cherchez une méthode qui "montre" quelque chose sur l'écran

DÉFI 1 : BONJOUR LE MONDE

- Imprimez "Hello World" sur votre matrice de lumière

```
>>> import hub
>>> hub.
__class__        __name__         __version__      BT_VCP
Image            USB_VCP          battery          ble
bluetooth        button           display          info
led              motion           port             power_off
sound            status           supervision      temperature
>>> hub.display.
__class__        callback         clear            pixel
rotation         show
>>> hub.display.show('Hello World')
```

GÉNÉRIQUE

- Cette leçon a été créée par Sanjay Seshan et Arvind Seshan pour « SPIKE Prime Lessons »
- D'autres leçons sont disponibles à l'adresse suivante www.primelessons.org